



Màster universitari en **Formació del Professorat d'Educació Secundària
Obligatòria i Batxillerat, Formació Professional i Ensenyament d'Idiomes**

Treball de fi de màster

Títol: L'ensenyament de l'algorítmia i la programació de computadors a l'assignatura de Tecnologia.

Cognoms: Calle Reus

Nom: Gerard

Titulació: Màster en Formació del Professorat d'Educació Secundària Obligatòria i Batxillerat, Formació Professional i Ensenyament d'Idiomes

Especialitat: Tecnologia

Director/a: Roser Cussó Calabuig

Data de lectura: 01/07/2015



«Gracias a la vida que me ha dado tanto
Me dio dos luceros que cuando los abro
Perfecto distingo lo negro del blanco
Y en el alto cielo su fondo estrellado
Y en las multitudes la mujer que yo amo.

Gracias a la vida que me ha dado tanto
Me ha dado el sonido y el abecedario
Con él las palabras que pienso y declaro
Madre amigo hermano y luz alumbrando,
La ruta del alma del que estoy amando.

Gracias a la vida que me ha dado tanto
Me ha dado la marcha de mis pies cansados
Con ellos anduve ciudades y charcos,
Playas y desiertos montañas y llanos
Y la casa tuya, tu calle y tu patio.

Gracias a la vida que me ha dado tanto
Me dio el corazón que agita su marco
Cuando miro el fruto del cerebro humano,
Cuando miro al bueno tan lejos del malo,
Cuando miro al fondo de tus ojos claros.

Gracias a la vida que me ha dado tanto
Me ha dado la risa y me ha dado el llanto,
Así yo distingo dicha de quebranto
Los dos materiales que forman mi canto
Y el canto de ustedes que es el mismo canto
Y el canto de todos que es mi propio canto.

Gracias a la vida»

Violeta Parra, 1966





Índex general

1	Introducció.....	4
2	Definició i context del problema.....	5
2.1	Què és l'algorítmia i la programació de computadors?.....	5
2.2	Per què ensenyar algorítmia i programació de computadors a l'educació secundària obligatòria?.....	7
2.3	L'ensenyament de l'algorítmia i la programació de computadors a l'educació secundària obligatòria en l'actualitat.....	9
2.3.1	Catalunya: Mobilitzem la informàtica.....	10
2.3.2	Catalunya: Impulsem la robòtica.....	11
2.3.3	Catalunya: Nou currículum de Tecnologia 2015/16.....	14
2.3.4	Madrid: "Programación Tecnología y robótica".....	14
2.3.5	Regne Unit: Currículum d'Informàtica.....	15
2.3.6	Estats Units: Code.org.....	17
3	Descripció de la solució.....	18
3.1	Com ensenyar algorítmia i programació de computadors a l'educació secundària obligatòria?.....	18
3.2	Objectiu d'aprenentatge: Entendre un algoritme.....	21
3.2.1	Expressar qualsevol procés mitjançant algoritmes.....	21
3.2.2	Trencaclosques.....	22
3.3	Objectiu d'aprenentatge: Detectar errors en un algoritme.....	23
3.3.1	Detectar errors semàntics en un algoritme.....	23
3.3.2	Detectar errors sintàctics en un algoritme.....	24
3.4	Objectiu d'aprenentatge: Construir un algoritme.....	25
3.4.1	Omplir forats.....	25
3.4.2	Treballar cooperativament.....	25
3.4.3	Crear algoritmes des de zero.....	26
3.5	Objectiu d'aprenentatge: Optimitzar un algoritme.....	26
3.5.1	Observa i millora.....	27
4	Resultats.....	28
4.1	Sessió a 2on de l'ESO.....	28
4.2	Sessió a 4rt de l'ESO.....	29
4.3	Lliçons apreses.....	30
5	Conclusions i treball a futur.....	31
6	Bibliografia.....	32
7	Webgrafia.....	34



Índex de figures

Figura 1 Algoritme descrit en al nivell que cerca si un número és parell o senar (Elaboració pròpia).....	6
Figura 2 Algoritme que cerca si un número és parell o senar expressat en pseudocodi (Elaboració pròpia).....	7
Figura 3 Algoritme que cerca si un número és parell o senar expressat en un diagrama de flux (Elaboració pròpia).....	7
Figura 4 Algoritme que cerca si un número és parell o senar expressat en el llenguatge de programació Javascript (Elaboració pròpia).....	8
Figura 5 Exemple de codi escrit en Logo per a dibuixar una cadira (Fotografia de Zhang Weiwu via Wikimedia Commons).....	9
Figura 6 AppInventor (Fotografia de Pùnsa via Wikimedia Commons).....	12
Figura 7 Placa Arduino Uno (Fotografia de SparkFun Electronics from Boulder, USA via Wikimedia Commons).....	13
Figura 8 Exemple de codi amb Processing per a programar una placa Arduino Uno (Elaboració pròpia).....	14
Figura 9 Interfaç de Scratch (Fotografia de Infofiltrage via Wikimedia Commons).....	18
Figura 10 Elements que determinen el grau de dificultat i assoliment de l'aprenentatge de l'algorítmia i programació de computadors. (Elaboració pròpia).....	19
Figura 11 Exemple d'algoritme explicant un procés de la vida quotidiana (Elaboració pròpia)...	22
Figura 12 Exemple conceptual d'una activitat de trencaclosques simple (Elaboració pròpia)...	23
Figura 13 Treball cooperatiu (Imatge de Scott Maxwell via Flickr).....	26
Figura 14 Funció que calcula si un número es parell o senar poc optimitzat (Elaboració pròpia).....	28



Índex de taules

Taula 1 Grau de dificultat segons l'objectiu d'aprenentatge, la tecnologia elegida i el tipus d'algoritme (Elaboració pròpia).....	20
Taula 2 Exemples d'activitats per a cadascun dels objectius d'aprenentatge (Elaboració pròpia).....	21
Taula 3 Algoritme que cerca si un número és parell o senar expressat en el llenguatge de programació Javascript amb errors semàntics (Elaboració pròpia).....	24
Taula 4 Algoritme que cerca si un número és parell o senar expressat en el llenguatge de programació Javascript amb errors sintàctics (Elaboració pròpia).....	25



Índex de figures

Taula 1 Algoritme descrit en al nivell que cerca si un número és parell o senar (Elaboració pròpia).....	6
Figura 2 Algoritme que cerca si un número és parell o senar expressat en pseudocodi (Elaboració pròpia).....	7
Figura 3 Algoritme que cerca si un número és parell o senar expressat en un diagrama de flux (Elaboració pròpia).....	7
Figura 4 Algoritme que cerca si un número és parell o senar expressat en el llenguatge de programació Javascript (Elaboració pròpia).....	8
Figura 5 Exemple de codi escrit en Logo per a dibuixar una cadira (Fotografia de Zhang Weiwu via Wikimedia Commons).....	9
Figura 6 AppInventor (Fotografia de Pùnsa via Wikimedia Commons).....	12
Figura 7 Placa Arduino Uno (Fotografia de SparkFun Electronics from Boulder, USA via	

1 Introducció

Des de fa uns anys hi ha un gran impuls per apropar l'ensenyament de l'algorítmia i la programació de computadors als instituts. Iniciatives mundials com code.org o l'arribada de tecnologies com Scratch, AppInventor o Arduino ho estan fent possible. Però per altra banda, s'obre un gran repte pel professorat de secundària i, especialment, pels professors de Tecnologia. Serem capaços d'ensenyar algorítmia o programació als nostres alumnes? I també molt important, coneixem els avantatges que ens aporta aquesta ciència i serem capaços de transmetre'ls als nostres alumnes?

Aquest treball final de màster, té com a objectiu principal crear un conjunt d'eines i activitats pel professorat de Tecnologia. Recursos que es puguin dur a l'aula i que integrin els diferents conceptes de l'algorítmia i la programació de computadors de forma transversal amb els continguts de l'assignatura de Tecnologia. Tot això, centrant-se específicament en els continguts curriculars de Tecnologia dels cursos de segon i tercer de l'Educació Secundària Obligatòria (ESO).

Primer s'analitzarà quins són els avantatges de l'ensenyament de l'algorítmia i la programació de computadors en els instituts i es reflexionarà sobre el perquè d'aquesta tendència sorgida els últims anys per a dur aquests coneixements a l'aula. És una necessitat de mercat o realment hi ha una creença que aquests coneixements ajuden als nostres alumnes a pensar, identificar problemes, cercar les solucions més òptimes o fomentar la seva creativitat?

Després s'estudiaran diferents casos reals, tant d'iniciatives a instituts com d'altres que s'han portat a través de la xarxa com poden ser code.org o codeacademy.com. En cadascuna d'elles s'estudiaran tant les seves metodologies com tecnologies utilitzades. És la gamificació una metodologia necessària per a apropar la programació als alumnes de secundària? També s'analitzaran quins són els entorns i canals possibles que es poden utilitzar per a fer arribar al professorat les activitats que es plantejaran a posteriori.

A partir de la recerca feta, es plantejarà una metodologia i tecnologies apropiades i es crearan tot un conjunt d'activitats i bones pràctiques per tal que el professorat de tecnologia les pugui adaptar fàcilment a les seves classes. Intentant sempre tenir en compte tota la diversitat d'alumnat, professorat i instituts que es troba a Catalunya. És possible ensenyar algorítmia en un institut sense un ordinador?

També es durà a terme una mostra de les activitats proposades en un entorn real amb l'objectiu d'analitzar quines són les fortaleeses i debilitats de la metodologia i tecnologia escollides. Per això, es coordinarà amb un professor de Tecnologia d'institut per a portar a l'aula aquesta experiència.

Per últim, s'extrauran i es sintetitzaran les conclusions de tot el treball realitzat i es comentarà el possible treball a futur que es podria realitzar en aquesta línia d'investigació.

2 Definició i context del problema

2.1 Què és l'algorítmia i la programació de computadors?

Conèixer les diferències entre els conceptes d'algorítmia i programació de computadors és clau per entendre com i perquè s'ha d'introduir aquest tipus d'ensenyament a l'educació secundària. També és important reconèixer les diferents maneres com podem expressar un algoritme o un llenguatge de computadors, per tal d'analitzar quina de les formes és la més adient a l'hora de proposar activitats als nostres alumnes. A continuació es fa una breu introducció a aquests conceptes i de quina manera estan interrelacionats entre ells.

Entenem l'algorítmia com la ciència que estudia els algoritmes [Brassard, Bratley i García-Bermejo, 1997]. Respecte als algoritmes o algorismes, paraula d'origen llatí i grec que prové del matemàtic persa Al-Juarismi, no existeix un consens clar sobre la seva definició. Segons el SEVOCAB (Software And Systems Engineering Vocabulary), un algoritme és la successió finita de regles precises per a la solució d'un problema en un nombre finit de passos¹. Però aquesta definició, no sempre es compleix i per això no és del tot acceptada. Per exemple, l'algoritme de Chudnovsky mai no aconsegueix trobar tots els decimals del número Pi [Chudnovsky, D. V., 1989].

Quan parlem de programació de computadors, ens referim a aquella activitat destinada al disseny, implementació, modificació i testeig de programes d'ordinadors². Aquests programes d'ordinadors són l'expressió formal d'un algoritme mitjançant un llenguatge de programació en concret.

Els algoritmes, per tant, es poden expressar com un llenguatge de programació i d'això se'n diu el nivell d'implementació. Però els algoritmes també es poden expressar d'altres formes [Sipser 1997]. En total, distingim els següents tres nivells:

- **Descripció d'alt nivell:** En aquest nivell s'estableix el problema, se selecciona un model matemàtic i es descriu l'algoritme de forma verbal i ometent tot tipus de detalls dels llenguatges de programació, com ara, la declaració de variables. A la Figura 1 es pot veure un exemple senzill d'un algoritme descrit en alt nivell. Aquest algoritme calcula si un número enter es tracta d'un número parell o senar:

Algoritme parellOSenar:
Per saber si un número és parell o senar, calcularem el residu d'aquest número dividit per dos. Si el residu és "0" el número serà parell, en cas contrari serà senar.

Figura 1 Algoritme descrit en al nivell que cerca si un número és parell o senar (Elaboració pròpia)

- **Descripció formal:** En aquest nivell, la descripció de l'algoritme és independent de qualsevol llenguatge de programació, però el seu format s'assembla a un llenguatge de programació d'alt nivell. S'explica l'algoritme pas a pas de manera verbal i utilitzant

¹ SEVOCAB: Software and Systems Engineering Vocabulary. Term: Algorithm. Obtingut 20 July 2015.

² SEVOCAB: Software and Systems Engineering Vocabulary. Term: Programming. Obtingut 20 July 2015.

llenguatges no formals o il·lustracions com, per exemple, el pseudocodi o els diagrames de flux.

El pseudocodi és la combinació del llenguatge natural amb les estructures del llenguatge de programació per a expressar el disseny d'un programa de computadors³. A la Figura 2, es mostra un exemple senzill d'un algoritme expressat en pseudocodi que, depenent el valor de l'entrada, ens indica si es tracta d'un número parell o senar:

```

Procés parellOSenar:
  Llegir numero
  Si (numero MOD 2 = 0) Llavors
    Escriure "És parell"
  Sino
    Escriure "És senar"
  FiSi
FiProcés
  
```

Figura 2 Algoritme que cerca si un número és parell o senar expressat en pseudocodi (Elaboració pròpia)

Un diagrama de flux és la representació gràfica d'un procés o la solució pas a pas d'un problema, utilitzant de forma adequada figures geomètriques connectades entre sí per línies de flux⁴. A la Figura 3, es mostra un altre cop l'algoritme que ens indica si un número és parell o senar però, aquesta vegada, expressat amb un diagrama de flux:

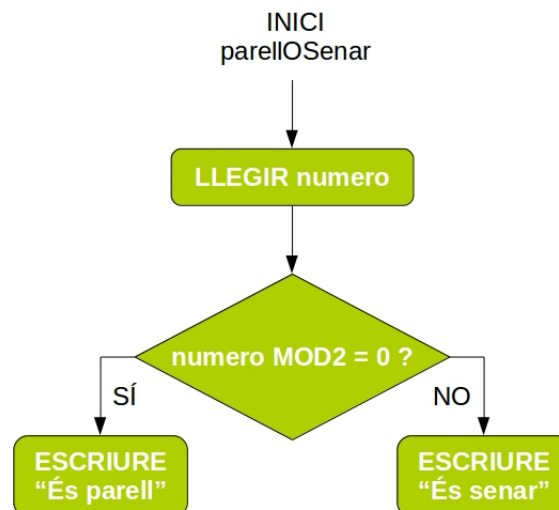


Figura 3 Algoritme que cerca si un número és parell o senar expressat en un diagrama de flux (Elaboració pròpia)

- **Implementació:** En aquest nivell, l'algoritme és expressat en un llenguatge de programació específic. En l'actualitat, existeixen multitud de llenguatges de programació depenent de la solució que estem cercant⁵. Per exemple, en la següent

3 SEVOCAB: Software and Systems Engineering Vocabulary. Term: Pseudocode. Obtingut 21 July 2015.

4 SEVOCAB: Software and Systems Engineering Vocabulary. Term: Flow chart. Obtingut 21 July 2015.

5 Scriptol.com,. 'List Of Programming Languages In Alphabetical Order'. N.p., 2015. Web. 21 Maig 2015.

Figura 4, es pot veure l'algoritme que calcula si un número és parell o senar expressat en el llenguatge de programació Javascript:

```
<script type="text/javascript">
    function parellOSenar() {
        var numero = prompt("Escriu un número enter");
        if(numero % 2 == 0) {
            alert("És parell");
        }
        else {
            alert("És senar");
        }
    }
</script>
```

Figura 4 Algoritme que cerca si un número és parell o senar expressat en el llenguatge de programació Javascript (Elaboració pròpia)

2.2 Per què ensenyar algorítmia i programació de computadors a l'educació secundària obligatoria?

Actualment trobem una gran tendència per promoure l'aprenentatge de la programació a les escoles. Per exemple, a escala mundial, trobem iniciatives com Code.org⁶ o CodeAcademy⁷. Code.org és una organització sense ànim de lucre recolçada i promoguda per grans empreses com Amazon, Apple, Dropbox, Facebook, Google, Microsoft o Yahoo⁸. L'objectiu d'aquestes iniciatives es centren en augmentar el nombre de persones, especialment dones, amb coneixements de programació de computadors per a suplir la futura necessitat de programadors i així donar un impuls a l'economia⁹.

A Catalunya, les carreres tècniques i, especialment, la carrera d'Informàtica ha experimentat una important davallada en el nombre d'alumnes matriculats els últims anys. A més, cada vegada, les empreses estan tenint més problemes per a cobrir ofertes de feines que requereixen una formació en el camp de les tecnologies de la informació i comunicació [Cussó i Gil, 2014]. Aquest fet obliga a reflexionar sobre la necessitat de cercar noves dinàmiques que facin que els alumnes, i especialment les dones, se sentin més atrets per les carreres STEM (Science, Technology, Engineering i mathematics).

A la Facultat d'Informàtica de Barcelona de la Universitat Politècnica de Catalunya (FIB-UPC), el percentatge de dones matriculades a Enginyeria Informàtica el curs 2008-09 no superava el 10% [Cussó, Fernández i Gil, 2009]. El curs 2013-14, a la Universitat de Washington, un 30% dels alumnes llicenciats en Informàtica l'any 2014 va ser dona. El perquè d'aquesta gran diferència entre les dues universitats, s'extreu d'un estudi on s'afirma que un dels motius és la introducció de l'ensenyament de l'algorítmia i la programació de computadors en els instituts dels Estats Units [Eney, Lazowska, Martin i Reges, 2013]. En aquesta línia, el Govern està establint noves mesures per adaptar la formació a les necessitats del mercat i així poder

6 Code.org., 'Every Child Deserves Opportunity'. N.p., 2015. Web. 22 Maig 2015.

7 Codecademy., 'Learn To Code'. N.p., 2015. Web. 22 Maig 2015.

8 Code.org., 'Partners'. N.p., 2015. Web. 22 Maig 2015.

9 Cadie Thompson, CNBC.com. 'Codecademy: Teach People To Code, Boost The Economy'. USA TODAY. N.p., 2015. Web. 22 Maig 2015.

competir en una economia globalitzada¹⁰. Aquestes mesures, passen per una revisió del currículum de Tecnologia, on a partir del pròxim curs 2014-15, s'inclourà l'ensenyament de l'algorítmia i la programació de computadors en els cursos de 2^{on} i 3^{er} de l'ESO¹¹ a Catalunya.

Malgrat aquestes iniciatives, marcades més per una necessitat de mercat, fa més de 40 anys que existeixen projectes que intenten incentivar l'aprenentatge de l'algorítmia i la programació de computadors amb motiu dels beneficis que s'obtenen desenvolupant un pensament computacional. El primer llenguatge de programació dissenyat amb fins didàctics va ser Logo¹², creat l'any 1967. La filosofia de Logo es basa en la teoria del Constructivisme de Piaget, on es concep l'aprenentatge com un procés on els alumnes creen coneixement a partir de la interacció amb altres persones o objectes del seu entorn¹³. A la següent Figura 5, es mostra un exemple del llenguatge de programació Logo dibuixant una cadira:

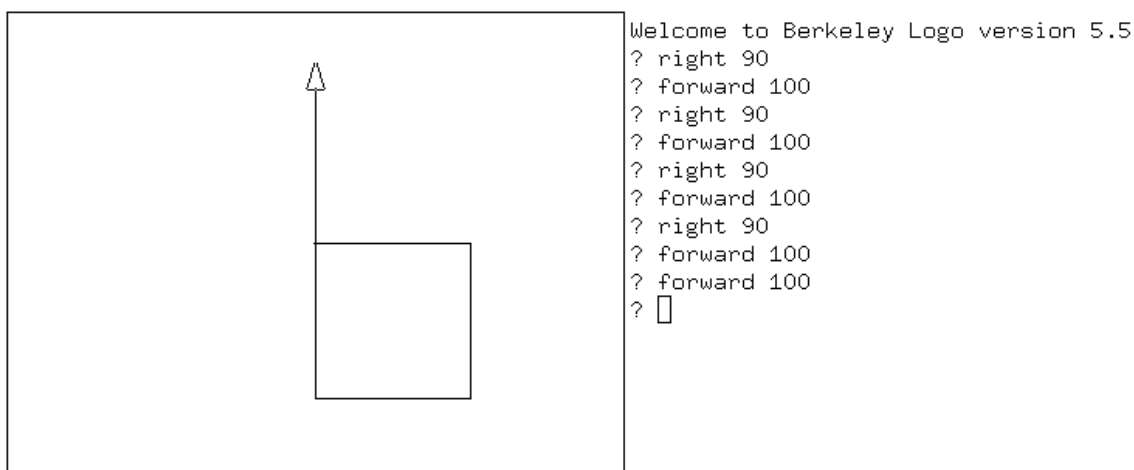


Figura 5 Exemple de codi escrit en Logo per a dibuixar una cadira (Fotografia de Zhang Weiwu via Wikimedia Commons)

Des de l'arribada de Logo al món de l'educació, trobem diversos estudis que demostren els avantatges de l'ensenyament de l'algorítmia i la programació de computadors. Un dels primers estudis va ser l'any 1973, anomenat "*The Effects of Computer Programming on Performance in Mathematics*" [Milner, 1973]. En aquest estudi s'afirma que els estudiants que treballen amb Logo mostren una millor facilitat per a resoldre problemes matemàtics. Més tard, a l'any 1984, l'estudi "*Effects of computer programming on young children's cognition*" [Clements, Gullo, 1984] afirmava que els estudiants que havien treballat amb Logo havien augmentat la seva capacitat per a produir idees creatives, resoldre problemes i ser més autònoms. Aquests investigacions s'han anat reafirmant amb altres estudis com "*The Effect of Logo Programming*

10 Press, Europa. 'Economía.- Educación Apuesta Por Adecuar La Formación De Los Universitarios A Las Necesidades Del Mercado Laboral'. europapress.es. N.p., 2015. Web. 22 Maig 2015.

11 Dogc.gencat.cat. 'EDICTE de 29 d'abril de 2015, pel qual se sotmet a informació pública el Projecte de decret d'ordenació dels ensenyaments de l'educació secundària obligatòria.'. N.p., 2015. Web. 22 Maig 2015.

12 El.media.mit.edu. 'Logo Foundation'. N.p., 2015. Web. 22 Maig 2015.

13 El.media.mit.edu. 'Logo And Learning'. N.p., 2015. Web. 22 Maig 2015.

Language for Creativity and Problem Solving [Pardamenan, Evelin, Honni, 2011], on també es va concloure que els alumnes que treballaven amb Logo eren més creatius i havien millorat la seva capacitat de resoldre problemes. També són importants les conclusions tretes d'altres estudis com *"Children and computers: New technology. Old concerns."* [Wartella i Jennings, 2000], on s'afirma que l'aprenentatge d'aquestes tecnologies ajuda a la socialització entre els companys i entre l'alumne i el professor.

Llavors podem observar que, malgrat la iniciativa dels últims anys per apropar l'ensenyament de l'algorítmia i la programació de computadors per una necessitat més de mercat, ja fa temps que es defensa el seu ús i la seva inclusió en l'educació secundària. El seu ensenyament es considera un element clau per a potenciar l'habilitat de resoldre de problemes, especialment matemàtics, la creativitat o la socialització entre companys i relacionant directament aquests beneficis amb les competències bàsiques que l'alumnat de l'ESO ha d'assolir en acabar el seu ensenyament obligatori¹⁴: Tractament de la informació i competència digital (CB3), competència matemàtica (CB4), competència d'aprendre a aprendre (CB5), competència d'autonomia i iniciativa personal (CB6) i competència social i ciutadana (CB8).

2.3 L'ensenyament de l'algorítmia i la programació de computadors a l'educació secundària obligatòria en l'actualitat

Elliot Soloway a l'any 1986 va analitzar com els llibres que ensenyaven programació se centraven en la sintaxi del llenguatge de programació. Però en canvi, va observar com els alumnes, que entenien perfectament aquests conceptes, tenien problemes per a resoldre els problemes que es plantejaven: *"Conèixer on posar el punt i coma en un programa no té per què portar a una millor solució del problema"* [Soloway, 1986].

Derek Sleeman, l'any 1986, defensava que el més important per aprendre a programar era adquirir les habilitats necessàries per a crear plans i mecanismes que resolguin problemes, fent ús, especialment, dels diagrames de flux [Sleeman, 1986]. No es defineix l'objectiu del programador com el coneixement de la sintaxi i semàntica del llenguatge de programació, sinó com l'assoliment de les següents habilitats:

1. Especificar un pla detallat i realitzable que solucioni un problema (Algoritme).
2. Implementar el pla en un llenguatge de programació.
3. Depurar el programa resultant.

Aquests eixos, ja establerts fa quasi 30 anys, haurien de ser la base de l'ensenyament de l'algorítmia i la programació de computadors a la secundària avui en dia. Però a més, durant aquests anys, on la societat i la tecnologia ha anat evolucionant, han sorgit altres elements que poden ajudar a l'aprenentatge d'aquest tipus d'ensenyament. Amb l'objectiu de trobar quines són les claus que determinen aquest èxit o fracàs i quins són els continguts, tecnologies i metodologies utilitzades, a continuació es descriuen i s'analitzen algunes iniciatives. Primer s'analitzaran dues propostes que s'han dut a terme a Catalunya, com són "Mobilitzem la informàtica" i "Impulsem la robòtica". A continuació, s'analitzaran els nous currículums que s'implantaràn l'any que ve a Catalunya i Madrid i a on s'inclourà l'ensenyament de l'algorítmia i

¹⁴ Currículum educació secundària obligatòria. Decret 143/2007 DOGC núm. 4915 Generalitat de Catalunya.

la programació de computadors. Seguidament s'estudiarà un dels casos pioners d'aquest ensenyament a Europa, el del Regne Unit. I per acabar, es comentarà com s'està duent a terme aquest ensenyament als Estats Units gràcies a iniciatives com Code.org.

2.3.1 Catalunya: Mobilitzem la informàtica.

Es tracta d'una proposta didàctica de l'assignatura optativa específica d'informàtica a 4^{rt} d'ESO iniciada el curs 2013-14. Segons es pot consultar a la pàgina Web de la XTEC, "l'eix fonamental d'aquesta proposta és un projecte col·laboratiu perquè l'alumnat dissenyi i desenvolupi una app per a dispositius mòbils"¹⁵. L'assignatura situa als alumnes com a membres d'una empresa, sent ells els encarregats de cercar una necessitat en el mercat, crear una solució per mitjà d'una App i, finalment, comercialitzar el producte resultant. Les claus d'aquesta proposta són:

- Recursos didàctics pel professorat en format Moodle¹⁶. El Moodle és una plataforma d'aprenentatge dissenyada per a proporcionar a educadors, administradors i estudiants un sistema integrat únic, robust i segur per a crear ambients d'aprenentatge personalitzats¹⁷. Des de la Generalitat de Catalunya, s'està donant un impuls per la implantació d'aquesta tecnologia amb plataformes com Àgora¹⁸ o Alexandria¹⁹. Àgora facilita un sistema d'allotjament als centres educatius oferint serveis com Moodle. Mentre que a Alexandria, trobem una biblioteca de recursos digitals per l'aula. Conté un conjunt de cursos Moodle, fàcilment importables, entre els quals estan els recursos de la proposta didàctica de "Mobilitzem la informàtica".
- Fer ús d'un Llenguatge de Programació Visual (LPV), l'AppInventor²⁰. Un LPV es caracteritza per utilitzar expressions gràfiques en el procés de construcció, modificació i execució de programes [Blackwell,1996]. El seu avantatge és que apropen el llenguatge de programació al nivell del disseny algorítmic, fent molt més fàcil entendre l'estructura del programa i eliminant els errors purament sintàctics del llenguatge. Però l'ús dels LPV sense una formació prèvia per adquirir les habilitats per a crear plans i mecanismes per a resoldre problemes (Algoritmes) pot dur al fet que l'estudiant no sigui capaç d'estructurar els blocs per a crear una solució del problema [Sleeman, 1986]. Altres problemes detectats són la seva escalabilitat, ja que és molt costós modificar un programa visual i l'ús ineficient que fan de l'espai disponible a la pantalla [Casares 1999].

AppInventor és un LPV creat per l'empresa Google i l'institut de Tecnologia de Massachusetts (MIT), amb l'objectiu de facilitar la programació d'aplicacions per a telèfons intel·ligents amb el sistema operatiu Android²¹. A la següent Figura 6, es pot veure l'entorn de programació d'AppInventor, el qual es pot executar directament en línia des d'un navegador Web sense la necessitat d'instal·lar-ho a l'ordinador. Això fa

15 Xtec.cat., 'XTEC - Currículum i Orientació. Educació Secundària Obligatoria. Optativa d'informàtica a 4rt d'ESO'. N.p., 2015. Web. 27 Maig 2015.

16 Moodle.org., 'Moodle - Open-Source Learning Platform | Moodle.Org'. N.p., 2015. Web. 28 Maig 2015.

17 Docs.moodle.org., 'Acerca De Moodle - Moodledocs'. N.p., 2015. Web. 28 Maig 2015.

18 Agora.xtec.cat., 'Àgora'. N.p., 2015. Web. 28 Maig 2015.

19 Alexandria.xtec.cat., 'Curs Informàtica 4rt d'ESO'. N.p., 2015. Web. 28 Maig 2015.

20 Appinventor.mit.edu., 'MIT App Inventor | Explore MIT App Inventor'. N.p., 2015. Web. 28 Maig 2015.

21 Appinventor.mit.edu., 'About Us | Explore MIT App Inventor'. N.p., 2015. Web. 28 Maig 2015.

que sigui possible treballar des de qualsevol ordinador amb sistemes operatius diferents.

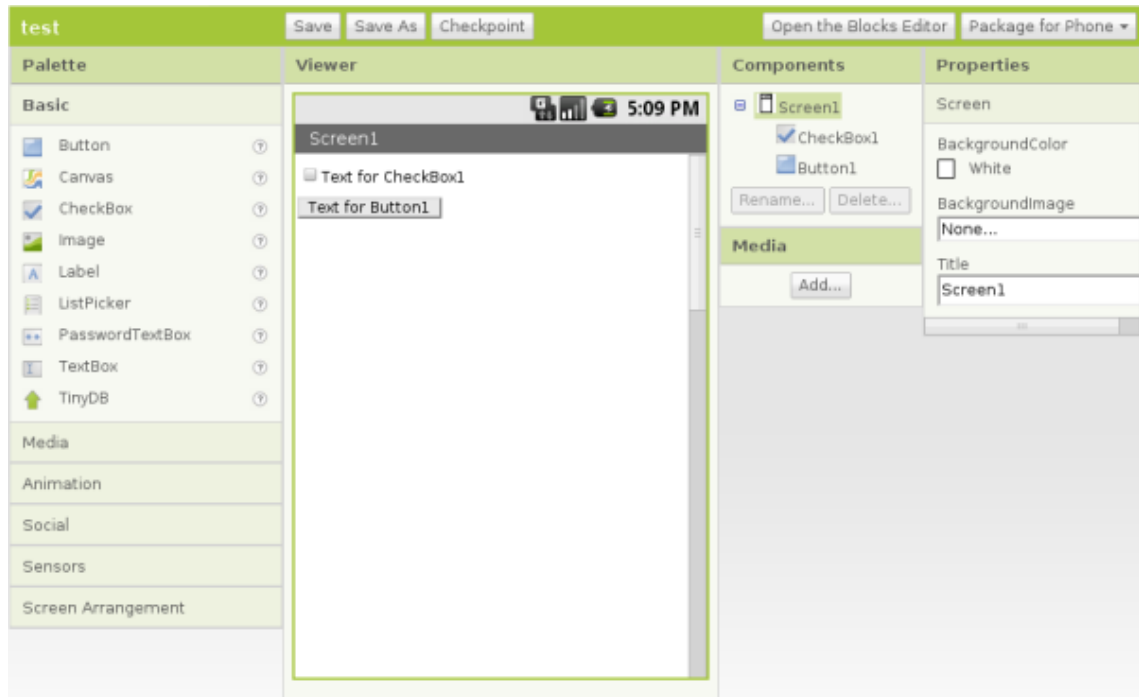


Figura 6 AppInventor (Fotografia de Pülsa via Wikimedia Commons)

- Un dels principals eixos de la proposta és la formació al professorat per garantir l'èxit de la seva implantació. Com comenta aquesta professora en l'article: "Quant al professorat, hem treballat la formació, perquè és essencial per poder comptar amb l'entusiasme dels docents"²². A més, s'ha de destacar la implicació de voluntaris externs a l'entorn escolar, que participen com a assessors experts i que ajuden a donar seguretat al professorat i a professionalitzar l'alumnat.
- Un concurs per a donar a conèixer els projectes creats per l'alumnat, compartir experiències entre centres i premiar a les millor propostes²³, incentivant així la implicació de l'alumnat, el professorat i els instituts.

2.3.2 Catalunya: Impulsem la robòtica.

Aquest projecte pilot, iniciat el curs 2014/15, s'ha impartit a uns 1.500 alumnes de 3^{er} i 4^{rt} d'ESO en un total de 52 centres de secundària. L'objectiu del projecte és "motivar els alumnes a explorar l'electrònica a través d'una sèrie de projectes de programació, construcció i

²² Upsocial.org., 'Promoure L'esperit Emprenedor Través De La Formació Tecnològica | Upsocial'. N.p., 2015. Web. 28 Maig 2015.

²³ Xtec.cat., 'XTEC - Centres. Als Centres... Premis I Concursos. Concurs D'apps 2014'. N.p., 2015. Web. 27 Maig 2015.

montatge”²⁴. Aquesta experiència es porta a terme gràcies a la placa Arduino²⁵, una placa de circuit imprès simple que té l'objectiu de fer més simple i accessible el disseny de circuits electrònics amb microcontroladors. A la Figura 7, podem veure una imatge d'una placa Arduino Uno.

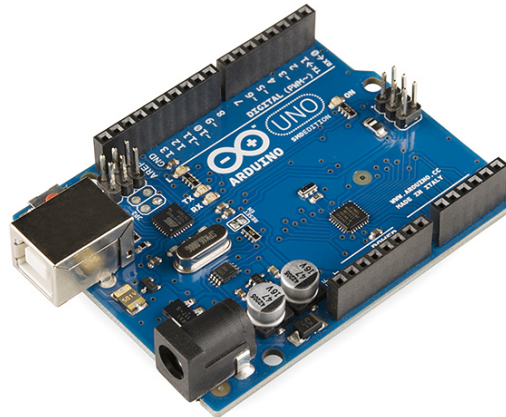


Figura 7 Placa Arduino Uno (Fotografia de SparkFun Electronics from Boulder, USA via Wikimedia Commons)

Dins dels continguts del curs, trobem una part dedicada a la programació del microcontrolador d'Arduino. A continuació, es detallen les claus d'aquest projecte i com es treballa l'ensenyament de la programació de microcontroladors amb l'alumnat:

- Creació d'una pàgina Web amb els continguts didàctics per l'alumnat²⁶. Aquest entorn proporciona una guia per l'alumnat, on pot anar seguint pas a pas cada una de les activitats proposades en el projecte. Però els continguts giren molt entorn la semàntica i la sintaxi del llenguatge, sense treballar conceptes algorítmics. Aquests conceptes s'haurien d'haver treballat prèviament si es vol que l'alumnat sigui capaç de superar el repte de resoldre nous problemes [Sleeman, 1986].
- Es troba a faltar una guia docent que ajudi al professorat a impartir i avaluar l'assignatura, així com podem trobar al Moodle del projecte “Impulse'm la informàtica”.
- En aquest cas, no es treballa amb cap LPV, sinó que l'alumnat treballa directament amb un llenguatge de programació textual (LPT) com és Processing²⁷. Aquest salt és un gran repte, tant per l'alumnat com pels professors, si els alumnes mai han treballat amb cap llenguatge de programació abans. Ja, que els alumnes, a part de treballar l'electrònica i cercar com resoldre problemes, hauran de fer front a un conjunt de

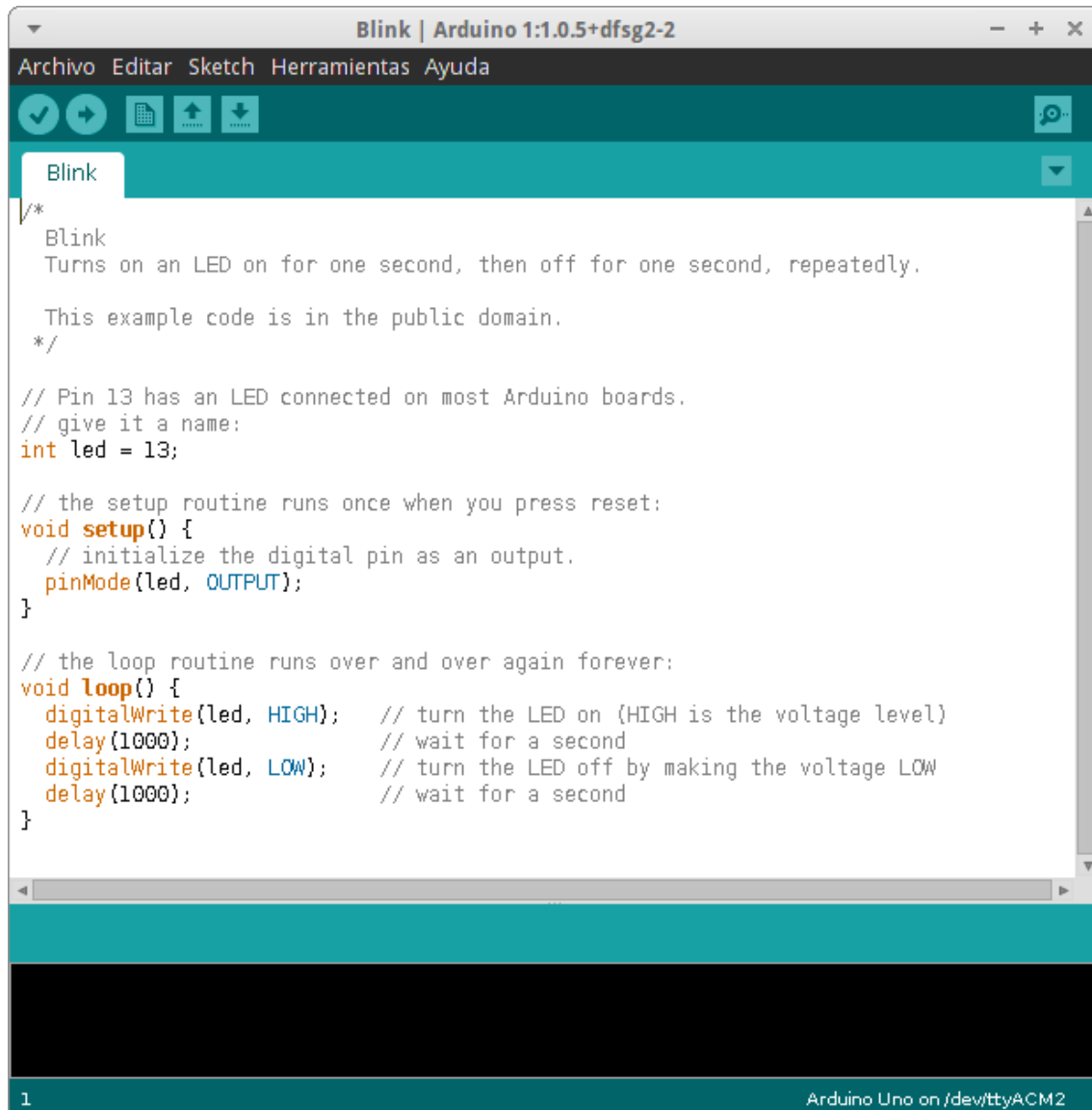
24 Premsa.gencat.cat. 'Més de 1.500 alumnes catalans aprenen a crear entorns interactius a través d'un nou projecte de robòtica i tecnologies creatives'. Sala De Premsa. Generalitat De Catalunya. N.p., 2015. Web. 28 Maig 2015.

25 Arduino.cc. 'Arduino - Home'. N.p., 2015. Web. 28 Maig 2015.

26 Bcn.verkstad.cc. 'Creative Technologies | Barcelona'. N.p., 2015. Web. 28 Maig 2015.

27 Processing.org. 'Processing.Org'. N.p., 2015. Web. 28 Maig 2015.

problemes semàntics i sintàctics propis del llenguatge [Soloway, 1986]. A la Figura 8, podem veure un exemple de codi amb Processing per a programar una placa Arduino Uno, on un Led, situat en el pin digital 13, va alternant el seu estat d'encès a apagat cada segon.



```

Blink | Arduino 1:1.0.5+dfsg2-2
Archivo Editar Sketch Herramientas Ayuda
Blink
/*
 * Blink
 * Turns on an LED on for one second, then off for one second, repeatedly.
 *
 * This example code is in the public domain.
 */

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}
1 Arduino Uno on /dev/ttyACM2

```

Figura 8 Exemple de codi amb Processing per a programar una placa Arduino Uno (Elaboració pròpia)

- Un projecte d'aquestes característiques fa que sigui necessària una important inversió, tant per a dotar els instituts dels components necessaris per a treballar com per a donar formació al professorat. En aquest sentit, cada centre ha estat dotat de kits d'iniciació a la robòtica que contenen tots els components per a realitzar i construir els experiments. A més uns 100 professors han estat formats prèviament per a poder impartir l'assignatura

i disposen d'una plataforma en línia amb tota la documentació necessària i un entorn web on compartir amb altres participants els continguts²⁸.

- Igual que en el projecte "Mobilitzem la informàtica", al final del curs es realitza una fira, amb l'objectiu de mostrar els projectes realitzats per l'alumnat, tant a la comunitat educativa com a la societat, i ser un estímul per l'alumnat implicat²⁹.

2.3.3 Catalunya: Nou currículum de Tecnologia 2015/16.

El pròxim curs 2015/16 entrarà en vigor el nou currículum de l'assignatura de Tecnologia a Catalunya³⁰. Un dels eixos principals serà l'ensenyament de l'algorítmia i la programació de computadors a partir de 2^{on} de l'ESO. Aquest fet, pot ser un impuls per assentar les bases d'iniciatives com "Mobilitzem la Informàtica" o "Impulsem la robòtica" comentades anteriorment. Ja es poden consultar els continguts i els criteris d'avaluació de l'assignatura de Tecnologia, el que ens pot permetre fer un breu anàlisi de quines seran les bases del seu ensenyament per determinar posteriorment quines poden ser les millors metodologies i les tecnologies a utilitzar.

El curs de 2^{on} de l'ESO se centra en l'algorítmia i a entendre les bases dels llenguatges de programació utilitzant LPV. Al curs de 3^{er} de l'ESO, es comença a treballar amb la implementació d'algoritmes amb LPT i la depuració de programes. Aquests continguts es relacionen perfectament amb les tres funcions que Sleeman defensa que ha de tenir un programador: Especificar un pla detallat i realitzable que solucioni un problema, implementar el pla en un llenguatge de programació i depurar el programa resultat [Sleeman, 1986]. A més, com es comentava anteriorment, començar a treballar amb un LPV és clau per ajudar a reforçar els conceptes algorítmics evitant entrar en les complexitats semàntiques i sintàctiques d'un LPT.

Un fet a destacar és que el nou currículum no s'enfoca a treballar amb cap tecnologia en concret, com ara la robòtica o els telèfons intel·ligents, sinó que considera aquest ensenyament com un eix transversal dins dels mateixos continguts de l'assignatura.

2.3.4 Madrid: "Programación Tecnología y robótica".

Durant el curs 2014/15, s'ha impartit per primera vegada l'assignatura de Programació a 4^{rt} de l'ESO a 15 instituts tecnològics de la Comunitat de Madrid³¹. Aquesta iniciativa pilot, ha estat una antesala del nou currículum que començarà a implantar-se el curs 2015/16 i on s'introduirà l'assignatura de "*Programación, Tecnología y Robótica*"³² a tots els nivells de l'ESO. Els continguts de la nova assignatura giraran entorn de la creació de pàgines Web, videojocs, aplicacions per a telèfons intel·ligents, la robòtica o la impressió 3D. Comparat amb el currículum que s'implantarà a Catalunya, trobem diferències destacades:

28 Edubcn.cat. 'L'institut Príncep de Viana acull la presentació d'un nou programa de robòtica i tecnologies creatives - Consorci d'educació de Barcelona'. 2014. Web. 28 Maig 2015.

29 Premsa.gencat.cat. 'Estudiants d'ESO de les comarques Barcelonines presenten 242 projectes de tecnologies creatives i robòtica a la fira de la tecnologia'. Sala De Premsa. Generalitat De Catalunya'. Web. 28 Maig 2015.

30 Xtec.cat. Currículum provisional 1r i 3r ESO 2015-2016. Proposta curricular de l'ESO. Àmbit científicotecnològic. Web. 28 Maig 2015.

31 Marcos, José. 'Los Colegios De Madrid Impartirán Clases De Programación'. EL PAÍS. N.p., 2015. Web. 26 Maig 2015.

32 Bcn.es. DECRETO 48/2015, de 14 de Maig, del Consejo de Gobierno, por el que se establece para la Comunidad de Madrid el currículo de la Educación Secundaria Obligatoria. Comunidad de Madrid, Consejería de Educación, Juventud y Deporte. Web. 26 Maig 2015.

- Els alumnes començaran a treballar continguts relacionats amb la programació a partir de 1^{er} de l'ESO fent ús dels LPV.
- Els continguts tenen un enfocament més pràctic que a Catalunya, donant molta importància a la robòtica i a la programació de dispositius mòbils des dels primers cursos.
- Malgrat que també s'estudia l'anàlisi i la resolució de problemes, no es comença fins a 2^{on} de l'ESO. Aquest ensenyament hauria de fer-se en el primer curs, conjuntament amb els LPV, ja és la base de l'aprenentatge de qualsevol llenguatge de programació, sigui visual o textual [Soloway, 1986].

Durant el primer any de prova pilot, aquesta assignatura ha obert certs debats dins de la comunitat educativa. La Comunitat de Madrid, conjuntament amb l'empresa Telefónica, han obert una pàgina Web³³ amb l'objectiu de fer front a la gran necessitat de professors formats en l'àmbit de la programació i la robòtica. El portal ofereix un curs de 50 hores pels professors que se suposa que seran suficients per a poder impartir l'assignatura. Malgrat això, hi ha una queixa pel poc temps de preparació que tindrà el professorat i es demana que la implantació de la nova assignatura sigui més gradual. L'altra queixa ve donada per la forta inversió que hauran de fer els centres en la compra de components electrònics per a impartir els continguts de robòtica, i on es demana una major inversió per part de la Comunitat de Madrid³⁴.

2.3.5 Regne Unit: Currículum d'Informàtica

El Regne Unit, juntament amb Estònia, ha estat un dels països pioners a la implantació de l'ensenyament de l'algorítmia i la programació de computadors tant a primària com a secundària, iniciant el nou currículum en el curs 2014/15. És important destacar aquest fet, ja que els alumnes del Regne Unit comencen a treballar el pensament computacional ja des de la primària. Centrant-nos en l'estudi de la programació a secundària, el Departament per l'Educació ha creat el portal "*Computing at school*"³⁵ per ajudar al professorat a introduir la programació en el currículum. El portal disposa d'una guia pels professors ("*Computing in the national curriculum. A guide for secondary teachers*"³⁶), espais de debat i recursos, com per exemple, un manual de programació en Python o un llibre amb petits exercicis per ajudar a fomentar el pensament computacional en l'alumnat ("*The Little Book of Programming Challenges*"³⁷).

La guia del professorat és una eina indispensable, ja que no només explica els continguts que s'han d'impartir sinó que ajuda als professors a elegir una metodologia i tecnologia adequada. Tot això, fent ús d'exemples molt clarificadors i entrant en detall, fins i tot, de com explicar algorismes de cerca o d'ordenació. Alguns aspectes a destacar de la guia relacionats amb la metodologia són els següents:

33 codemadrid.org, 'Code Madrid'. N.p., 2015. Web. 29 Maig 2015.

34 eldiario.es, 'La Asignatura De Programación En Madrid Es Un 'Copia-Pega' De Otras Materias'. N.p., 2014. Web. 29 Maig 2015.

35 Computingatschool.org.uk, 'Computing At School: Promoting And Supporting Excellence In Computer Science Education'. N.p., 2015. Web. 31 Maig 2015.

36 Computingatschool.org.uk, 'Computing At School :: Secondary Computing'. N.p., 2015. Web. 31 Maig 2015.

37 www.iwebsolutions.co.uk, iWeb. 'The Little Book Of Programming Challenges - National STEM Centre'. Nationalstemcentre.org.uk. N.p., 2015. Web. 31 Maig 2015.

- La imaginació és el límit i els continguts es poden adaptar a qualsevol context. Per això és important explicar les bases del pensament computacional de manera transversal, incloent-hi aquest ensenyament en altres continguts del currículum e inclús involucrant altres assignatures.
- L'ensenyament s'ha de centrar a aprendre algorítmia i programació, no en com utilitzar la tecnologia.
- És important planificar adequadament els cursos, de tal manera que permeti que els alumnes vagin progressant fins a assolir els continguts del currículum.
- S'ha d'intentar centrar l'ensenyament en resultats i eines que motivin a l'alumnat i que siguin propers a ells.
- L'ensenyament de la programació sense ordinadors pot ajudar als alumnes a entendre millor certs conceptes que després poden ser reforçats per mitjà de l'ordinador.
- No és estrany que un programador dediqui un 50% del seu temps a corregir errors. Per això, és important ensenyar als programadors novells tècniques per a detectar i corregir errors més ràpidament.

La guia del professorat també pretén ajudar als docents a elegir quina tecnologia és la més adient per a cada cas. Per això, demana que els professors es facin aquestes tres preguntes:

- Com de bé domino el llenguatge de programació?
- Existeixen recursos de qualitat i una comunitat per a donar suport a l'aprenentatge i l'ensenyament d'aquest llenguatge?
- Poden accedir als alumnes fàcilment des de casa i l'escola a aquest llenguatge?

Aquestes preguntes haurien de ser la base per a l'elecció de la tecnologia a utilitzar en cada institut. Però a més, es fa un repàs pels diferents llenguatges de programació que es podrien portar a l'aula:

- Començar amb un LPV com Scratch³⁸ o Kodu³⁹ és un excel·lent punt de partida per a assentar les bases del pensament computacional, però també, l'alumne hauria d'aprendre un LPT. Scratch és el LPV més utilitzat arreu del món, està basat en el llenguatge de programació LOGO i el seu objectiu és ser una eina que serveixi per aprendre a programar a través del joc i l'experimentació. A la Figura 9, es pot veure la interfície de Scratch, la qual pot ser executada directament en línia des de qualsevol navegador.
- Quant als LPT, es destaquen diversos llenguatges com Java, Visual Basic, Python o Javascript. A més, es remarca la importància dels llenguatges de programació associats a fulls de càlcul.

38 Scratch.mit.edu., 'Scratch - Imagine, Program, Share'. N.p., 2015. Web. 31 Maig 2015.

39 Kodugamelab.com., 'Kodu | Home'. N.p., 2015. Web. 31 Maig 2015.

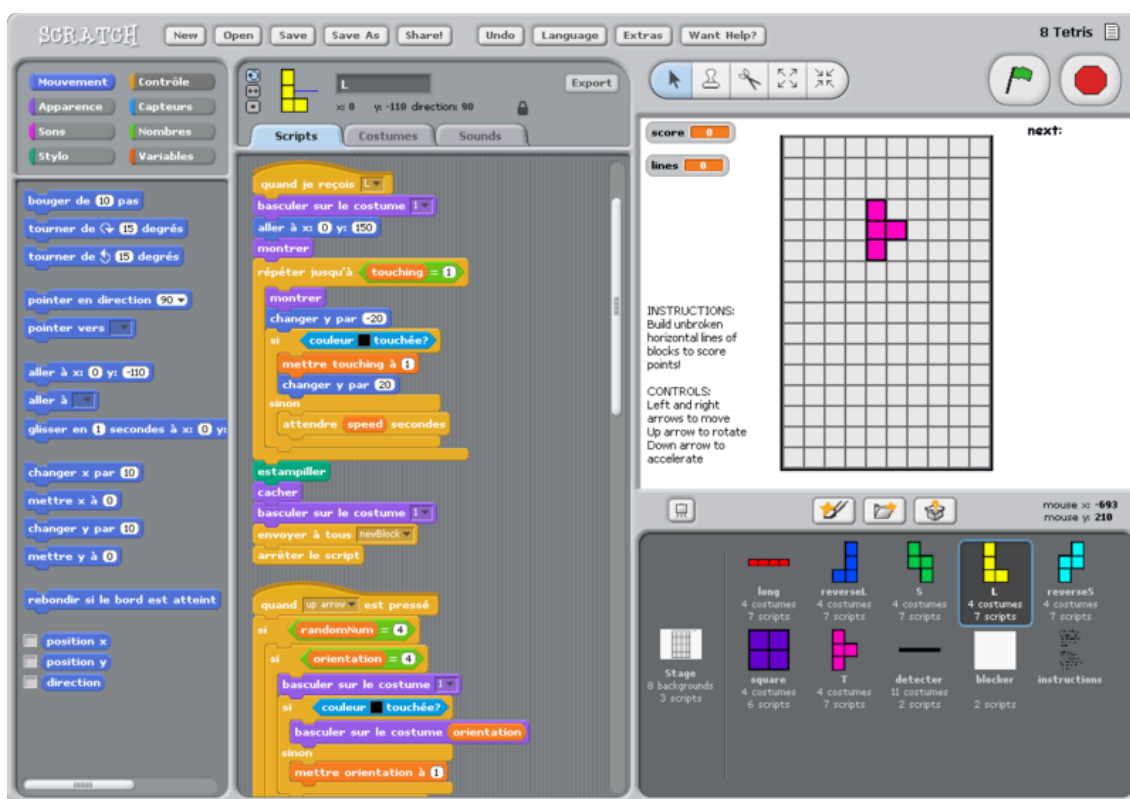


Figura 9 Interfaç de Scratch (Fotografia de Infolitrage via Wikimedia Commons)

2.3.6 Estats Units: Code.org

El projecte Code.org ha estat impulsat per grans companyies tecnològiques amb l'objectiu d'introduir l'algorítmia i la programació de computadors a l'educació secundària. Dins de la seva metodologia, trobem moltes característiques semblants a les comentades anteriorment, com per exemple, l'aprenentatge de l'algorítmia sense utilitzar ordinadors o fer ús d'un LPV. Però hi ha un element significatiu que representa una gran diferència i és la incorporació de la gamificació.

La gamificació o ludificació consisteix en “aplicar estratègies de jocs en contextos aliens als dels jocs, amb l'objectiu que les persones adoptin certs comportaments” [Ramírez, 2014]. Per exemple, en el cas de Code.org, a mesura que els alumnes van avançant en el seu ensenyament, van rebent punts i trofeus que van marcant el seu grau d'aprenentatge. Aquesta tendència en l'educació està molt lligada al gran increment d'hores que passen els joves d'avui en dia jugant a videojocs, ja que es creu que utilitzant les mecàniques que usen els jocs dels videojocs pot ajudar a fomentar una sana competició entre l'alumnat [Cortizo, 2011]. Malgrat això també hi ha detractors de les teories de la gamificació, ja que es considera que condueixen cap a un infantilisme de l'educació i que redueixen la motivació intrínseca per l'aprenentatge [Pink, 2009].

3 Descripció de la solució

3.1 Com ensenyar algorítmia i programació de computadors a l'educació secundària obligatòria?

Casos com els de Madrid o Regne Unit, estan sent qüestionats per una falta de recursos o per una poca formació del professorat en la matèria^{40,41}. Però per altra banda, projectes com el de "Mobilitzem la informàtica" demostren que, disposar d'un material⁴² ben preparat tant per l'alumnat com pel professorat, ajuden al fet que iniciatives d'aquest tipus siguin un èxit. A partir de l'estudi fet en el punt anterior "2.3. L'ensenyament de l'algorítmia i la programació de computadors a l'educació secundària obligatòria en l'actualitat", es pot afirmar que l'ensenyament de l'algorítmia i la programació de computadors a la secundària ha d'estar precedit per tres fets inqüestionables per a garantir el seu èxit:

- Una correcta formació del professorat.
- Una forta inversió que asseguri que els instituts tinguin els recursos necessaris
- La creació d'un banc de recursos, guies o fòrums on els professors puguin recolzar-se per a impartir les seves classes.

Aquest treball de màster se centra en l'última de les tres línies comentades anteriorment, la de proporcionar als professors de tecnologia d'un conjunt de bones pràctiques i activitats amb l'objectiu que siguin un suport per les seves classes.

L'objectiu d'aprenentatge a assolir per part de l'alumnat de secundària, referent a l'ensenyament de l'algorítmia i la programació de computadors, ha d'estar marcat per les bases ja preestablertes per Sleeman l'any 1986: els alumnes han de ser capaços d'entendre un algorisme, detectar errors, construir els seus propis algorismes i, per últim, optimitzar per fer-los més eficients. Per arribar a aquest objectiu, és important tenir en compte dos elements que també determinen el grau de dificultat i assoliment d'aquest aprenentatge: la tecnologia i el tipus d'algorisme.

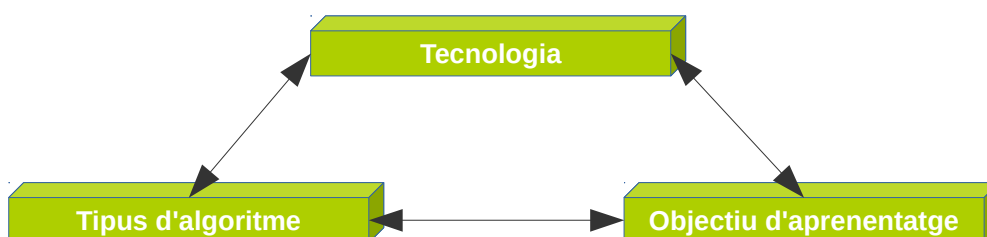


Figura 10 Elements que determinen el grau de dificultat i assoliment de l'aprenentatge de l'algorítmia i programació de computadors. (Elaboració pròpia)

40 eldiario.es., 'La Asignatura De Programación En Madrid Es Un 'Copia-Pega' De Otras Materias'. N.p., 2014. Web. 29 Maig 2015.

41 BBC News., 'Are Teachers Ready For The Coding Revolution? - BBC News'. N.p., 2015. Web. 14 Juny 2015.

42 Alexandria.xtec.cat., 'Curs Informàtica 4rt d'ESO'. N.p., 2015. Web. 28 Maig 2015.

Per a iniciar als alumnes en el món del pensament computacional, una de les millors maneres és fer-ho sense utilitzar ordinadors i amb l'ajuda d'eines com els diagrames de flux, evitant així les distraccions i els problemes tècnics derivats de treballar amb ordinadors [Bell, Witten i Fellows 2015]. Els conceptes adquirits es poden anar reforçant amb l'ajuda de LPV, incorporant l'element tecnològic en l'ensenyament però treballant amb un llenguatge pròxim a l'alumnat. Un cop assolida una base, es pot fer el salt a un LPT que s'adapti a les limitacions i necessitats del centre i l'alumnat. En aquesta fase s'ha de tenir molt present tots els problemes derivats de la sintàctica i semàntica del llenguatge i recolzar a l'alumnat perquè no decaigui la seva motivació. I per finalitzar, es pot orientar la programació a un caràcter més pràctic i motivador, però a la vegada més exigent, com pot ser la robòtica o la programació de telèfons intel·ligents.

Els algoritmes també tenen un grau de dificultat depenent del procés que descriguin. El recomanable és partir d'algoritmes lineals, és a dir, que comencin i acabin sense que el seu flux es vegi afectat per cap iteració o condicional. I de mica en mica, anar incorporant elements que el facin més complex com variables, càlculs matemàtics, condicionals o iteracions. Per últim, l'alumnat hauria de treballar amb algoritmes complexos com els de cerca o ordenació.

A partir de les diferents activitats que es proposen en els casos que s'han anat comentant i analitzant en aquest treball (Mobilitzem la informàtica, Code.org, CS Unplugged...), s'han creat les següents taules, amb l'objectiu que siguin una guia que reflecteixi el grau de dificultat que pot tenir una activitat segons els criteris comentats anteriorment: l'objectiu d'aprenentatge, la tecnologia elegida i el tipus d'algoritme.

Grau de dificultat segons l'objectiu d'aprenentatge

Nivell 1: Entendre un algoritme.

Nivell 2: Detectar errors en un algoritme.

Nivell 3: Construir un algoritme.

Nivell 4: Optimitzar un algoritme.

Grau de dificultat segons la tecnologia elegida

Nivell 1: Sense fer ús d'ordinadors.

Nivell 2: Llenguatges de Programació Visual.

Nivell 3: Llenguatges de Programació Textual.

Nivell 4: Robòtica i Programació de telèfons intel·ligents.

Grau de dificultat segons el tipus d'algoritme

Nivell 1: Algoritme lineal.

Nivell 2: Elements d'entrada i sortida.

Nivell 3: Constants, variables i operadors aritmètics.

Nivell 4: Estructura condicional.

Nivell 5: Estructura iteració.

Nivell 6: Algoritmes complexos (Cerca i ordenació).

Taula 1

Grau de dificultat segons l'objectiu d'aprenentatge, la tecnologia elegida i el tipus d'algoritme
(Elaboració pròpia)

En aquest treball, les activitats se centraran en dos llenguatges de programació: Scratch (LPV) i Javascript (LPT). Malgrat això, totes les activitats poden ser extrapolades a qualsevol altre llenguatge de programació. Per elegir aquests llenguatges s'ha seguit la guia que es proporciona als professors de secundària al Regne Unit: són llenguatges de programació que domino, existeixen una gran quantitat de recursos de qualitat, hi ha una comunitat que dona suport a l'aprenentatge i ensenyament d'aquests llenguatges i els alumnes poden accedir fàcilment des de casa i l'escola, ja que no es necessita ni d'accés a Internet, ni d'ordinador amb grans recursos ni d'un sistema operatiu en concret. A més, els dos llenguatges es troben entre la llista de llenguatges recomanats per la guia del professorat de secundària del Regne Unit i tenen molta similitud amb dos llenguatges que poden ser treballats posteriorment a 4^{rt} de l'ESO: AppInventor i Processing.

Les activitats proposades giren entorn els continguts de l'assignatura de Tecnologia. Tenint en compte el currículum del curs 2015/16 a Catalunya, aquesta guia se centra en els cursos de 2^{on} i 3^{er} de l'ESO, que és on s'han d'introduir els principis de l'algorítmia i la programació de computadors. És per això que s'ha decidit deixar de banda la programació orientada a la robòtica o als telèfons intel·ligents, ja que en aquests cursos no es treballa en l'àmbit curricular a Catalunya. També s'han incorporat metodologies com la gamificació, però s'ha decidit que no sigui un eix de l'ensenyament, ja que pot conduir cap a un infantilisme de l'educació i a reduir la motivació intrínseca de l'alumnat per aprendre.

A continuació es descriuen una sèrie de bones pràctiques i activitats que poden servir de referència o punt de partida pel professorat de Tecnologia de secundària per tal d'introduir l'ensenyament de l'algorítmia i la programació de computadors a les seves classes. A la següent taula, s'indiquen les diferents activitats proposades per a cadascun dels objectius d'aprenentatge comentats anteriorment en forma de resum:

Objectiu d'aprenentatge: Entendre un algoritme

Expressar qualsevol procés mitjançant algoritmes.
Trencaclosques.

Objectiu d'aprenentatge: Detectar errors en un algoritme

Detectar errors sintàctics en un algoritme.
Detectar errors semàntics en un algoritme.

Objectiu d'aprenentatge: Construir un algoritme

Omplir forats.
Treball cooperatiu.
Construir algoritmes des de zero.

Objectiu d'aprenentatge: Optimitzar un algoritme

Observa i millora.

Taula 2

Exemples d'activitats per a cadascun dels objectius d'aprenentatge (Elaboració pròpia)

3.2 Objectiu d'aprenentatge: Entendre un algoritme

En aquest objectiu d'aprenentatge, s'ha d'intentar que els alumnes, a partir de la lectura d'un algoritme, siguin capaços de seguir el seu flux d'execució i descriure quin és el procés que realitza. Per aconseguir-ho es proposen les següents bones pràctiques i activitats:

3.2.1 Expressar qualsevol procés mitjançant algorismes

Amb l'objectiu que l'alumnat comenci a familiaritzar-se amb l'algorítmia i la programació de computadors, és indispensable que el llenguatge algorítmic estigui present a les aules. S'ha de tenir present que, qualsevol procés, inclús els de la nostra vida quotidiana, poden ser expressats mitjançant un diagrama de flux. Per exemple, a la figura 11 es pot observar un procés fictici que segueix un professor d'ençà que surt de casa seva fins que arriba a l'aula expressat amb un diagrama de flux.

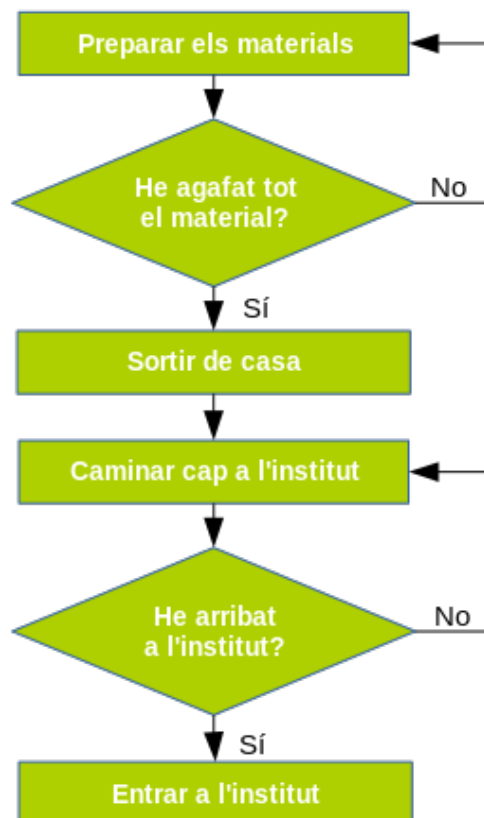


Figura 11 Exemple d'algoritme explicant un procés de la vida quotidiana (Elaboració pròpia)

Són molts els processos que es descriuen a l'assignatura de tecnologia, un d'ells i que és totalment transversal a tots els cursos de l'ESO, és la descripció del procés tecnològic. Aquest, pot ser explicat de forma descriptiva, o bé fent ús dels diagrames de flux com es pot observar a

l'annex "1. El procés tecnològic". Anar apropant aquest tipus de llenguatge a l'aula és el primer pas que s'ha de seguir per tal que l'alumnat sigui capaç de crear els seus propis algoritmes. Però no només es poden explicar continguts, una activitat per l'aula en si mateixa també pot ser expressada en format algorítmic. A l'annex "2. Joc "Sí o No" es mostra un exemple d'una activitat relacionada amb les fonts d'energia que és explicada mitjançant algoritmes.

3.2.2 Trencaclosques

Aquest tipus d'activitat són una primera presa de contacte abans que l'alumnat comenci a crear els seus propis algoritmes. En aquestes activitats, es donen totes les parts que formen l'algoritme en forma de blocs i els alumnes són els que han d'ordenar-les i assignar-los una lògica d'execució perquè l'algoritme faci el que s'ha marcat com a objectiu inicialment. A la figura 12, es pot veure un exemple conceptual d'una activitat de trencaclosques simple. On a l'esquerra tenim els blocs desordenats, que serà la informació que rebrà l'alumnat a l'inici i, a la dreta, l'objectiu a aconseguir que és ordenar aquests blocs per a donar-los una lògica d'execució.

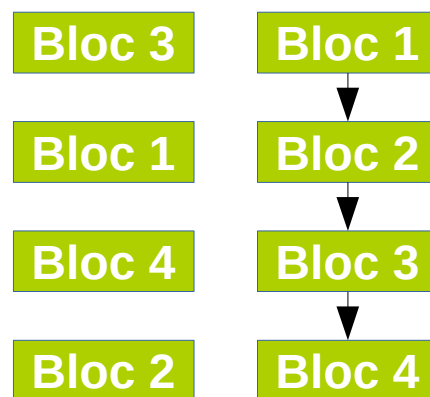


Figura 12 Exemple conceptual d'una activitat de trencaclosques simple (Elaboració pròpia)

Aquestes activitats presenten diversos avantatges:

- Es poden realitzar tant amb com sense ordinador.
- Són un bon punt de partida per a introduir a l'alumnat a Scratch, ja que utilitza un LPV representat per blocs.
- Es pot anar augmentant el seu grau de dificultat progressivament, afegint de mica en mica altres elements més complexos com condicionals, iteracions, variables, elements d'entrada i sortida o càlculs.
- Les peces es poden agrupar o descompondre inicialment, per tal de donar un grau de dificultat major o menor i així atendre a la diversitat de l'alumnat.

A l'annex "3 Activitat: El tèxtil" es mostra una activitat de tipus trencaclosques per alumnes de 2^{on} d'ESO fent ús de Scratch i utilitzant com a contingut el procés tèxtil.

3.3 Objectiu d'aprenentatge: Detectar errors en un algoritme

Com es comentava en la guia del professorat del Regne Unit, no és estrany que un programador dediqui un 50% del seu temps a corregir errors. En aquest sentit, és important incorporar activitats que ajudin a l'alumnat a detectar errors per reduir la seva frustració quan es trobin davant d'aquests problemes.

En aquest objectiu d'aprenentatge, els alumnes han de ser capaços d'interpretar un algoritme i detectar i corregir possibles errors. En el món de l'algorítmia i la programació de computadors, es distingeixen dos tipus d'errors:

3.3.1 Detectar errors semàntics en un algoritme

Són aquells errors que, encara que el compilador no els detecta, el programa no realitza la funció esperada. En general, són els errors més difícils de detectar i corregir. Aquests errors es poden donar en tots els nivells, ja que són independents de la tecnologia que s'usa. Per tant, fins i tot un algoritme expressat en un diagrama de flux pot presentar errors semàntics. Per exemple, a la Taula 3, tenim un algoritme en Javascript que cerca si un número és parell o senar amb un error semàntic:

1.	<script type="text/javascript">
2.	function parellOSenar() {
3.	var numero = prompt("Escriu un número enter");
4.	if(numero % 2 == 0) {
5.	alert("El número" + numero + "és senar");
6.	}
7.	else {
8.	alert("El número" + numero + "és parell");
9.	}
10.	}
11.	</script>

Taula 3 Algoritme que cerca si un número és parell o senar expressat en el llenguatge de programació Javascript amb errors semàntics (Elaboració pròpia)

Aquest programa donaria la solució a l'inrevés, quan el número fos parell donaria que és senar i quan el número fos senar donaria que és parell i, a més, el compilador no ho detectaria com un error al tractar-se d'un error semàntic. Podríem corregir-ho modificant la línia 4 de codi de la següent manera:

```
if(numero % 2 != 0) {
```

A l'annex "4. Joc Encerca i passa" es mostra una activitat d'exemple on es practica la detecció d'errors semàntics sense fer ús d'ordinadors.

3.3.2 Detectar errors sintàctics en un algorisme

Són aquells errors que es donen quan el codi està mal escrit i el compilador no és capaç d'interpretar-ho. Estan directament vinculats amb els LPT, ja que els LPV estan dissenyats per evitar aquests tipus d'errors. Aquests errors són molt diversos, ja que depenen del LPT amb el que s'estigui treballant i, per tant, el mateix tipus d'error, a vegades pot implicar reaccions diferents segons el llenguatge. Per exemple, a la Taula 4, tenim un algorisme en Javascript que cerca si un número és parell o senar amb diversos errors sintàctics freqüents:

1.	<script type="text/javascript">
2.	function parellOSenar() {
3.	numero = prompt("Escriu un número enter");
4.	if(numero % 2 == 0) {
5.	alert(EI número + numero + "és parell");
6.	}
7.	else {
8.	alert("El número" + numero + "és senar")
9.	}
10.	}
11.	</script>

Taula 4 Algorisme que cerca si un número és parell o senar expressat en el llenguatge de programació Javascript amb errors sintàctics (Elaboració pròpia)

Els errors que es troben en l'algorisme anterior són:

- Línia 3: Utilització d'una variable sense haver-la declarat prèviament. Correcció:

var numero = prompt("Escriu un número enter");

- Línia 5: Error de concatenació de cadenes de caràcters. Correcció:

alert("El número" + numero + "és parell");

- Línia 8: Error de final d'instrucció. Correcció:

alert("El número" + numero + "és senar");

A l'annex "5 Activitat: Les palanques" es mostra una activitat d'exemple on es practica la detecció d'errors sintàctics i semàntics amb el LPT Javascript.

3.4 Objectiu d'aprenentatge: Construir un algoritme

Malgrat que molts projectes i iniciatives analitzades anteriorment centren l'ensenyament de l'algorítmia i la programació de computadors entorn de la creació d'aplicacions, la metodologia que es presenta en aquest treball la incorpora com a un objectiu més dels quatre que ha d'assolir l'alumnat.

En aquest objectiu d'aprenentatge, els alumnes han de ser capaços de, a partir de les seves pròpies idees, plasmar-les en un algoritme. Per aconseguir aquest objectiu es presenten tres tipus d'activitats, cadascuna amb un nivell de dificultat major:

3.4.1 Omplir forats

Aquest tipus d'activitats són una primera pressa de contacte en la construcció d'algoritmes. Als estudiants se'ls dona la major part del codi resolt i ells han d'anar omplint petites instruccions. Aquests tipus d'activitats presenten diversos avantatges:

- Són una bona eina per a practicar la sintàctica dels LPT.
- Es poden realitzar tant amb com sense ordinador, però en general són de gran ajuda per tal que l'estudiant comenci a crear algoritmes tant amb LPV com LPT.
- Es pot anar augmentant el seu grau de dificultat progressivament modificant la mida dels forats.

A l'annex "6 Activitat: Magnituds elèctriques" es mostra una activitat d'exemple on s'han d'anar omplint diferents forats de programes fets amb Scratch per a calcular el valor de la intensitat en circuits elèctrics.

3.4.2 Treballar cooperativament

En el món laboral, quan es treballa la programació de computadors, un dels elements claus és el treball cooperatiu. Moltes vegades és impossible que un programador realitzi ell sol tot un programa, per tant, el que es fa és dividir-lo en petites parts i repartir-les entre tots els programadors.



Figura 13 Treball cooperatiu (imatge de Scott Maxwell via Flickr)

En aquest tipus d'activitats, no només es vol mostrar aquesta manera de treballar sinó que també es vol incentivar la capacitat de l'alumnat per a treballar en equip, desenvolupar projectes en comú o gestionar de forma positiva conflictes.

Per a realitzar aquestes activitats, cada alumne assumeix una petita tasca. Només si tots els membres del grup realitzen la seva part correctament, el programa funcionarà. A l'annex "7 Activitat: Els engranatges" es mostra una activitat d'exemple on és necessari el treball cooperatiu per tal que el programa funcioni correctament. L'activitat es realitza amb el LPT Javascript i el marc teòric dels engranatges i els seus càlculs associats.

3.4.3 Crear algorismes des de zero

Aquest tipus d'activitats són l'objectiu final a aconseguir en aquest nivell, però per aconseguir-ho, abans requereix un llarg aprenentatge: Entendre algorismes, saber detectar i corregir errors i omplir parts d'algorismes.

En aquest tipus d'activitats, entren en joc dos elements fonamentals: la creativitat i la capacitat de l'alumnat per transcriure un procés en un algorisme. Creativitat per a pensar una idea i capacitat per transcriure aquesta idea en un conjunt d'instruccions que ordenades amb la lògica adequada i precisa, descriguin el seu comportament.

A l'annex "8 Activitat: Motxilla energètica" es mostra una activitat d'exemple on es practica la creació d'algorismes a partir d'idees pròpies de l'alumnat. L'activitat gira entorn de l'estalvi energètic, on els alumnes hauran de pensar un mecanisme que els ajudi a estalviar energia a la seva llar i descriure el seu funcionament mitjançant un algorisme.

3.5 Objectiu d'aprenentatge: Optimitzar un algorisme

Tant en el món de la informàtica com en el nostre dia a dia, realitzar les nostres activitats de la forma més eficient possible és una de les claus d'èxit que ens permet estalviar recursos i esforços. Per exemple, una empresa de transports ha de decidir cada dia quina és la ruta més òptima per tal que realitzi les seves entregues de la forma més eficient. Això li permet estalviar costos en benzina, en transportistes i que els paquets arribin més aviat.

Quan l'alumnat treballa l'algorítmia i la programació de computadors s'ha de tenir molt present l'optimització. Que un programa realitzi la seva funció, no implica que ho estigui fent de la forma més eficient possible. Per exemple, a la figura 14 podem veure un algorisme escrit en Javascript que realitza un càlcul per saber si un número és parell o senar, però ho fa de forma molt poc eficient. El motiu és el nombre total d'instruccions que executa, comparat amb l'algorisme que es va presentar a la figura 4. L'algorisme de la figura 4 executa sempre 3 instruccions, mentre que l'algorisme de la figura 14 executa $X + 3$ instruccions, sent X el número que es vol calcular si és parell o senar.

```
<script>
function parellOSenar() {
    var numero = prompt("Escriu un número enter");
    while (numero > 0){
        numero = numero - 2;
    }
    if(numero == 0) {
        alert("El número" + numero + "és parell");
    }
    else {
        alert("El número" + numero + "és senar");
    }
}
</script>
```

Figura 14 Funció que calcula si un número es parell o senar poc optimitzat (Elaboració pròpia)

Malgrat que hi ha altres elements com l'espai que es consumeix en memòria, o el tipus d'instrucció, aquests factors s'han descartat ja que es considera que són massa complexos pel nivell de secundària.

Per a practicar aquest objectiu d'aprenentatge, es proposa el següent tipus d'activitat:

3.5.1 Observa i millora

En aquestes activitats, es presenta als alumnes un algoritme no optimitzat i han de ser capaços de detectar perquè no ho és i proposar millores per tal que realitzi la seva funció d'una manera més eficient. A l'annex "9 Activitat: Tipus d'esforços" es mostra una activitat d'exemple on es practica l'optimització d'un algoritme de cerca sense fer ús d'ordinadors i utilitzant com a marc teòric els diferents tipus d'esforços que pateixen objectes quotidians.

4 Resultats

Amb l'objectiu d'analitzar la resposta de l'alumnat davant el tipus d'activitats plantejades en el punt anterior, s'han realitzat 3 sessions en un institut de secundària. Aquestes sessions s'han realitzat en els cursos de 2^{on} (1 sessió) i 4^{rt} de l'ESO (2 sessions). Malgrat que els cursos objectiu de les activitats plantejades anteriorment són 2^{on} i 3^{er}, s'ha decidit realitzar dues sessions a 4^{rt} d'ESO. El motiu és que, encara que la major part dels alumnes de 4^{rt} d'ESO ja havien treballat la programació de computadors amb Scratch i orientada a la robòtica i als telèfons intel·ligents amb els projectes de “Mobilitzem la informàtica” i “Impulsem la robòtica”, presentaven dificultats a l'hora de crear els seus propis algoritmes. Aquest fet, s'ha observat sobretot durant el seguiment que s'ha fet a l'alumnat de 4^{rt} d'ESO en el projecte de “Impulsem la robòtica”, on els alumnes no eren capaços de traslladar les seves idees al LPT Processing. A continuació es descriuen les sessions realitzades a l'institut amb els alumnes i els resultats obtinguts:

4.1 Sessió a 2^{on} de l'ESO

Per realitzar aquesta sessió es van establir tres requisits previs:

- Les activitats havien de girar entorn dels continguts que s'estaven treballant en les sessions anteriors. En aquest cas, les activitats havien de girar entorn de les diferents formes de generació d'energia.
- Les activitats plantejades havien de treballar els quatre objectius d'aprenentatge establerts en el punt anterior: entendre un algoritme, detectar errors en un algoritme, construir un algoritme i optimitzar un algoritme. Així es podria detectar en quin dels quatre objectius els alumnes presentaven més dificultats.
- Les activitats s'havien de poder realitzar sense ordinadors, per tal d'evitar qualsevol distracció o problema tècnic derivat del seu ús.

Per aquests motius, es va decidir portar a l'aula aquestes dues activitats:

1. Joc encerta i passa: Malgrat que el joc se centra en la detecció d'errors en un algoritme, incorpora de manera indirecta dos objectius d'aprenentatge més: entendre un algoritme i construir un algoritme. Els alumnes han d'entendre l'algoritme per anar seguint la dinàmica del joc i han de construir una part de l'algoritme per a corregir l'error detectat.
2. Es va modificar l'activitat comentada a l'annex “9. Tipus d'esforços” per tal de simplificar-la i contextualitzar-la amb els continguts que s'estaven treballant, amb l'objectiu de treballar també l'altre objectiu d'aprenentatge: optimitzar un algoritme. En comptes de demanar un algoritme expressat en un diagrama de flux, es va demanar simplement que ho descriguessin textualment i, en comptes de treballar amb tipus d'esforços, es va modificar per treballar amb fonts d'energia renovables i no renovables. A l'annex “10 Activitat d'optimitzar un algoritme modificada”, es pot veure la descripció de l'activitat modificada així com els diferents elements que contenen els

sobres i el seu ordre.

Durant la primera activitat, la participació i implicació de l'alumnat va ser molt alta durant tot el joc. Van ser capaços d'entendre i seguir l'algoritme sabent en cada moment l'estat de les variables i com modificar-les. A l'hora de detectar l'error, malgrat que sabien que la variable "X" s'havia de tornar a inicialitzar, no van aconseguir donar un algoritme que funcionés correctament. A l'annex "11 Resposta dels alumnes en el joc Encerta i passa" es mostra la resposta que els alumnes majoritàriament van donar. Aquesta resposta, malgrat que torna a inicialitzar la variable, dóna un error, ja que poden tornar a jugar grups que ja estan eliminats. Malgrat això, s'ha de tenir en compte el grau de dificultat que té resoldre aquest error i que el simple fet de proposar una possible solució, encara que no sigui correcta, ja té molt de mèrit en alumnes que no han treballat mai l'algorítmia.

La segona activitat d'optimització d'un algoritme suposava un gran repte pels alumnes, no només per la dificultat en si, sinó també per la falta de temps que es disposava (10 minuts). S'ha de dir que la majoria dels grups es van sentir frustrats i incapaços de donar una resposta vàlida i només un grup va ser capaç de respondre correctament. Malgrat això, el fet que un grup d'alumnes de 2^{on} d'ESO sense experiència en algorítmia sigui capaç de, només en 10 minuts, d'optimitzar un algoritme, ja és una gran fita. La resposta que van donar realitza l'algoritme de cerca de dreta a esquerra, fent que sigui més complicat expressar-ho en un diagrama de flux, però igual d'optimitzat que l'algoritme que va d'esquerra a dreta. A l'annex "12 Respostes dels alumnes en l'activitat d'optimització d'un algoritme" es poden llegir les diferents respostes que els alumnes van donar per aquesta activitat.

S'ha observat que malgrat que alguns estudiants són capaços de realitzar algoritmes complexos sense experiència prèvia, es necessita anar treballant de mica en mica els diferents objectius d'aprenentatge marcats en aquest treball per evitar la frustració del grup classe. Un altre factor important a destacar, és que els alumnes es sorprenen quan al final se'ls comenta que aquest tipus d'exercicis són la base per a començar a fer programació de computadors, ja que els consideren més com un joc que com una activitat de programació. També el professor de tecnologia present a l'aula va remarcar l'interessant que era realitzar activitats d'aquest tipus, ja que els alumnes treballaven el pensament computacional de manera transversal als continguts de tecnologia i sense que els alumnes s'adonessin compte.

4.2 Sessió a 4^{rt} de l'ESO

Per realitzar aquesta sessió es van establir tres requisits previs:

- Les activitats havien d'estar contextualitzades tenint en compte els continguts que s'estaven treballant en les sessions anteriors. En aquest cas, els alumnes estaven treballant els continguts de la unitat didàctica d'habitatge.
- L'objectiu d'aprenentatge d'aquesta sessió havia de girar entorn de la construcció d'algoritmes. Per tal de treballar les mancances que s'havien detectat prèviament en el projecte d'"Impulsem la robòtica".
- Ja que els alumnes havien presentat dificultats en implementar algoritmes amb Processing, es volia evitar els errors derivats de treballar amb un LPT o un LPV. Per

això, es va decidir plantejar les activitats per realitzar-les sense ordinadors.

Com es comentava anteriorment, es van impartir dues sessions portant a l'aula l'activitat comentada a l'annex "8 Activitat: Motxilla energètica". Aquesta activitat compleix perfectament amb els requisits indicats anteriorment i s'emmarca dins de la unitat didàctica d'habitatge, fent que els alumnes treballin continguts relacionats amb la domòtica.

Primer de tot i amb l'objectiu d'apropar el llenguatge algorítmic usat en els diagrames de flux, en entrar a classe es va començar a descriure tot el procés que seguia un professor d'ença que sortia de casa seva fins que entrava a l'aula. Això va donar com a resultat un diagrama de flux molt similar al de la figura 11. Aquesta introducció va permetre veure que els estudiants entenen perfectament el flux d'un programa i va ser un punt de partida per a introduir l'activitat a fer durant les següents sessions.

Les dues sessions van resultar molt profitoses i els alumnes van participar activament realitzant les seves propostes i creant els diagrames de flux. Es va poder detectar les dificultats que tenien els alumnes, inclús els més brillants, en expressar les seves idees en un algoritme. Això evidencia el fet que, començar a programar directament amb telèfons intel·ligents o amb robots, no és una bona idea si abans no s'ha treballat prèviament els conceptes bàsics d'algorítmia i programació de computadors. A l'annex "13 Resposta d'un grup d'alumnes en l'activitat de la motxilla energètica" es pot llegir la proposta d'un grup d'alumnes, el seu diagrama de flux abans de la correcció, el diagrama de flux corregit i el diagrama de flux resultant. Com es pot observar en l'últim diagrama de flux, malgrat les correccions, encara presenta errors greus. Per exemple, en el moment que la temperatura sigui inferior a 25°C, mai més es tornarà a mirar la temperatura del sensor.

4.3 Lliçons apreses

Malgrat que l'experiència ha estat breu i el nombre total d'alumnes que han participat d'ella no és significatiu ni representatiu del que és l'alumnat a Catalunya, es poden treure les següents conclusions:

- L'ensenyament d'un llenguatge de programació no implica que l'alumnat hagi adquirit les habilitats necessàries per dominar l'algorítmia i la programació de computadors.
- L'ensenyament de l'algorítmia i la programació de computadors s'ha d'anar introduint de mica en mica, partint per reptes més senzills i anant augmentant el seu nivell progressivament, evitant així la frustració o desmotivació d'una gran part de l'alumnat.
- Dinàmiques com el joc "Encerca i passa" són una bona eina per introduir a l'alumnat en el món del pensament computacional. L'alumnat ho pren com un joc, però realment està practicant conceptes relacionats amb l'algorítmia.
- En línies generals, els alumnes tenen facilitat per entendre un algoritme, però els hi suposa un gran repte quan han de transcriure un procés a un diagrama de flux o a un llenguatge informàtic.

5 Conclusions i treball a futur

Durant els últims anys hi ha una gran tendència per part dels governs per incorporar en els currículums de secundària l'ensenyament de l'algorítmia i la programació de computadors. Aquesta iniciativa, sorgeix per a suplir la futura necessitat de programadors i així donar un impuls a l'economia dels països. Malgrat això, fa més de 40 anys que es defensa el seu ensenyament considerant-lo com un element clau per a potenciar l'habilitat de resoldre de problemes, especialment matemàtics, la creativitat o la socialització entre companys.

L'element clau per aprendre a programar no és aprendre un llenguatge de programació, sinó adquirir les habilitats necessàries per a crear plans i mecanismes que resolguin problemes. Moltes de les iniciatives que s'han portat actualment a l'educació secundària no incorporen aquesta filosofia, sinó que centren l'aprenentatge més en la tecnologia, com pot ser la robòtica o la programació de telèfons intel·ligents. L'arribada del nou currículum de secundària és una oportunitat per incorporar aquesta filosofia a l'hora d'aprendre un llenguatge de programació i, a més, convertir-la un eix transversal amb la resta de continguts de l'assignatura de tecnologia. Però abans d'incorporar l'ensenyament de l'algorítmia i la programació de computadors a la secundària, són necessaris tres fets inqüestionables: una correcta formació del professorat, una forta inversió que assegurí que els instituts disposen dels recursos necessaris i la creació d'un banc de recursos on els professors es puguin recolzar per a impartir les seves classes. Sense alguns d'aquests tres factors, es corre el risc de fracassar a l'hora d'apropar aquest ensenyament a l'alumnat provocant-li frustració i desmotivació.

Els objectius d'aprenentatge que ha d'assolir l'alumnat de secundària, referent a l'ensenyament de l'algorítmia i la programació de computadors, són: entendre un algorisme, detectar errors en un algorisme, construir els seus propis algorismes i, per últim, optimitzar els algorismes per fer-los més eficients. Per aconseguir-ho, també s'ha de tenir present dos elements més que també condicionen el grau de dificultat i assoliment d'aquest aprenentatge: la tecnologia escollida i el tipus d'algorisme. Per això, es creu necessari començar l'ensenyament de l'algorítmia i la programació de computadors sense fer ús d'ordinadors, evitant així tots els problemes tècnics i distraccions derivats d'ells. I de mica en mica, anar incorporant tecnologies més complexes, com ara els LPV, els LPT i finalment la robòtica i la programació de telèfons intel·ligents.

Tenint en compte els tres elements comentats anteriorment (objectius d'aprenentatge, tecnologia i tipus d'algorisme), s'han proporcionat diferents activitats i bones pràctiques per a cadascun dels objectius d'aprenentatge que poden ser adaptades i traslladades a l'aula. Per fer-ho, és necessari tenir present la diversitat de l'alumnat així com les limitacions del centre, del professorat, de l'alumnat i de la pròpia tecnologia elegida.

Quan al treball a futur, hi han diversos punts pendents a treballar:

- Crear un banc de recursos en format Moodle per tal que el professorat el pugui descarregar i començar a utilitzar directament.
- Crear més activitats i més diverses per tal que el professorat disposi de més recursos.
- Realitzar una comparació amb altres LPT, com per exemple Python o Java, per tal de veure realment quin pot ser més vàlid per aplicar a la secundària.



- Realitzar una mostra en els instituts molt més gran i exhaustiva per tal de poder extreure unes conclusions més profundes.

6 Bibliografia

- BELL, Tim, WITTEN, Ian H. i FELLOWS Mike (2015). CS Unplugged, An enrichment and extension programme for primary-aged students.
- BRASSARD, Gilles, BRATLEY, Paul i GARCIA-BERMEJO, Rafael (1997). Fundamentos De Algoritmia. Madrid [etc.]: Pearson Prentice Hall.
- BLACKWELL, A. (1996). Programming in pictures, pictures of programs. Discussion paper for the thinking with diagrams. Portsmouth, Reino Unido.
- CASARES, Juan Pablo (1999). AMIVA: Ambiente para la instrucción visual de algoritmos. instituto Tecnológico Autónomo de México. Mexico, D.F.
- CHUDNOVSKY, D.V. (1989). The Computation Of Classical Constants. Proceedings of the National Academy of Sciences 86.21: 8178-8182. DOI:10.1073/pnas.86.21.8178.
- CLEMENTS, D. H. i GULLO, D. F. (1984). Effects of computer programming on young children's cognition. Journal of Educational Psychology, 76, p. 1051-1058.
- CORTIZO, Jose Carlos (2011). Gamificación y Docencia: Lo que la Universidad tiene que aprender de los Videojuegos. VIII Jornadas Internacionales de Innovación Universitaria.
- CUSSÓ, Roser, GIL, Marisa (2014). Una Visió actual de l'estat de la dona en les enginyeries. En revista de tecnologia [en línia], p. 31-37. <<http://www.raco.cat/index.php/RevistaTecnologia/article/view/284787/372653>> [consulta: 22-05-15]
- CUSSÓ, Roser, FERNÁNDEZ, M.J., GIL, Marisa (2009). ICT4GIRLS Compartiendo Experiencias de Tecnología de la Información y la Comunicación (TIC) desde Secundaria a la Universidad. International Review of Engineering Education. 2, p. 1940-1116.
- ENEY, C., LAZOWSKA, E., MARTIN, H., REGES, S. (2013). Broadening participation: The why and the how. Computer, vol. 46, no. 3, p. 48-51.
- GALLEGO, F., MOLINA R. i LLORENS, F. (2014). Gamificar una propuesta docente. Diseñando experiencias positivas de aprendizaje. Dpto. de Ciencia de la Computación e Inteligencia Artificial Universidad de Alicante.
- MILNER, S. (1973). The Effects of Computer Programming on Performance in Mathematics. Web. 23 Maig 2015..
- PARDAMEAN, Bens, EVELIN, Evelin, HONNI, Honni (2011). The effect of logo programming language for creativity and problem solving. En Proceedings of the 10th WSEAS international conference on E-Activities. World Scientific and Engineering Academy and Society (WSEAS), p. 151-156.
- PINK, Daniel (2009). Drive: the surprising truth about what motivates us. Riverhead.
- RAMÍREZ, José Luis (2014). GAMIFICACIÓN. Mecánicas de juegos en tu vida personal y profesional. Servicio Comercial del Libro. Madrid, España.
- SIPSER, Michael (1997). Introduction To The Theory Of Computation. Boston: PWS Pub. Co.
- SLEEMAN, Derek (1986), The Challenges of Teaching Computer Programming. Communications of the ACM, Volume 29, Number 9.
- SOLOWAY, E. (1986). Learning to Program = Learning to Construct Mechanisms and Explanations. Communications of the ACM, Volume 29, Number 9.
- WARTELLA, E. A., & JENNINGS, N. (2000). Children and computers: New technology. Old concerns. The future of children, p. 31-43.



7 Webgrafia

- Agora.xtec.cat,. 'Àgora'. N.p., 2015. Web. 28 Maig 2015.
- Alexandria.xtec.cat,. 'Curs Informàtica 4rt d'ESO'. N.p., 2015. Web. 28 Maig 2015.
- Appinventor.mit.edu,. 'MIT App Inventor | Explore MIT App Inventor'. N.p., 2015. Web. 28 Maig 2015.
- Arduino.cc,. 'Arduino - Home'. N.p., 2015. Web. 28 Maig 2015.
- Areatecnologia.com,. 'JUEGOS DE TECNOLOGIA GRATIS'. N.p., 2015. Web. 29 Maig 2015.
- BBC News,. 'Are Teachers Ready For The Coding Revolution? - BBC News'. N.p., 2015. Web. 28 Maig 2015.
- Bcn.verkstad.cc,. 'Creative Technologies | Barcelona'. N.p., 2015. Web. 28 Maig 2015.
- Blog.educalab.es,. 'La Programación Informática Como Herramienta Didáctica | Blog De INTEF'. N.p., 2014. Web. 28 Maig 2015.
- Bocm.es. DECRETO 48/2015, de 14 de Maig, del Consejo de Gobierno, por el que se establece para la Comunidad de Madrid el currículo de la Educación Secundaria Obligatoria. Comunidad de Madrid, Consejería de Educación, Juventud y Deporte. Web. 26 Maig 2015.
- Cadie Thompson, CNBC.com. 'Codecademy: Teach People To Code, Boost The Economy'. USA TODAY. N.p., 2015. Web. 22 Maig 2015.
- Clic.xtec.cat,. 'Zonaclí - Català'. N.p., 2015. Web. 29 Maig 2015.
- Codecademy.com. 'Learn To Code'. N.p., 2015. Web. 22 Maig 2015.
- Codemadrid.org,. 'Code Madrid'. N.p., 2015. Web. 29 Maig 2015.
- Code.org. 'Every Child Deserves Opportunity'. N.p., 2015. Web. 22 Maig 2015.
- Computingatschool.org.uk,. 'Computing At School: Promoting And Supporting Excellence In Computer Science Education'. N.p., 2015. Web. 31 Maig 2015.
- Computingatschool.org.uk,. 'Computing At School :: Secondary Computing'. N.p., 2015. Web. 31 Maig 2015.
- DeAmicis, Carmel et al. 'By September Coding Will Be Mandatory In British Schools. What The Hell, America?'. PandoDaily. N.p., 2014. Web. 28 Maig 2015.
- Dogc.gencat.cat,. 'EDICTE de 29 d'abril de 2015, pel qual se sotmet a informació pública el Projecte de decret d'ordenació dels ensenyaments de l'educació secundària obligatòria.'. N.p., 2015. Web. 22 Maig 2015.
- Educn.cat,. 'L'institut Príncep de Viana acull la presentació d'un nou programa de robòtica i tecnologies creatives - Consorci d'educació de Barcelona'. 2014. Web. 28 Maig 2015
- Educativa, Explorador. 'Nuevo Currículum: "Programación" En El Aula'. Fundación telefónica. N.p., 2014. Web. 29 Maig 2015.
- Eldiario.es,. 'La Asignatura De Programación En Madrid Es Un 'Copia-Pega' De Otras Materias'. N.p., 2014. Web. 29 Maig 2015.
- El.media.mit.edu. 'Logo Foundation'. N.p., 2015. Web. 22 Maig 2015
- Emanuele Feronato,. 'Bar Balance Game Prototype Made In Under 4 Minutes With Construct2, Featuring Revolute Joints'. N.p., 2013. Web. 28 Maig 2015.
- Espeso, Pablo. 'Niños Y Programación: Consejos Y Recursos Para Que Este Verano Se Inicien'. Xataka.com. N.p., 2015. Web. 29 Maig 2015.
- Finley, Klint. 'Estonia Reprograms First Graders As Web Coders | WIRED'. WIRED. N.p., 2015. Web. 28 Maig 2015.



- Gardiner, Beth. 'Adding Coding To The Curriculum'. Nytimes.com. N.p., 2014. Web. 23 Maig 2015.
- Genbetadev.com,. "'La Enseñanza Del Pensamiento Algorítmico Debe Empezar En Primaria" Entrevista A Juan Julián Merelo'. N.p., 2015. Web. 29 Maig 2015.
- Hernández, P. 'Como Enseñar Programación Para Niños Y Jóvenes Recomendado | Profesores Digitales'. Profesoresdigitales.com. N.p., 2014. Web. 28 Maig 2015.
- Iwebsolutions.co.uk, iWeb. 'The Little Book Of Programming Challenges - National STEM Centre'. Nationalstemcentre.org.uk. N.p., 2015. Web. 31 Maig 2015.
- Kodugamelab.com,. 'Kodu | Home'. N.p., 2015. Web. 31 Maig 2015.
- Marcos, José. 'Los Colegios De Madrid Impartirán Clases De Programación'. EL PAÍS. N.p., 2015. Web. 26 Maig 2015.
- Martí, Jordi. 'Introducción A La Programación En La ESO | Xarxatic'. Xarxatic.com. N.p., 2014. Web. 29 Maig 2015.
- Mateos, Manu. 'Programación Y Educación: Qué Países La Tienen En Su Plan De Estudios'. T.genbeta.com. N.p., 2014. Web. 29 Maig 2015.
- Moodle.org,. 'Moodle - Open-Source Learning Platform | Moodle.Org'. N.p., 2015. Web. 28 Maig 2015.
- Pascal.computer.org. 'SEVOCAB: Software and Systems Engineering Vocabulary '. N.p., 2015. Web. 20 Julio 2015.
- Perezpastor.blogspot.com.es,. 'Tecnología Y TIC: 1º ESO'. N.p., 2015. Web. 28 Maig 2015.
- Prensa.gencat.cat. 'Estudiants d'ESO de les comarques Barcelonines presenten 242 projectes de tecnologies creatives i robòtica a la fira de la tecnologia'. Sala De Premsa. Generalitat De Catalunya'. Web. 28 Maig 2015.
- Prensa.gencat.cat. 'Més de 1.500 alumnes catalans aprenen a crear entorns interactius a través d'un nou projecte de robòtica i tecnologies creatives'. Sala De Premsa. Generalitat De Catalunya. N.p., 2015. Web. 28 Maig 2015.
- Press, Europa. 'Economía.- Educación Apuesta Por Adecuar La Formación De Los Universitarios A Las Necesidades Del Mercado Laboral'. europapress.es. N.p., 2015. Web. 22 Maig 2015.
- Processing.org,. 'Processing.Org'. N.p., 2015. Web. 28 Maig 2015.
- Programamos,. 'Programamos'. N.p., 2015. Web. 28 Maig 2015.
- Scratch.mit.edu,. 'Scratch - Imagine, Program, Share'. N.p., 2015. Web. 31 Maig 2015.
- Scriptol.com. 'List Of Programming Languages In Alphabetical Order'. N.p., 2015. Web. 21 Maig 2015.
- The Economist,. 'A Is For Algorithm'. N.p., 2014. Web. 29 Maig 2015.
- Upsocial.org,. 'Promoure L'esperit Emprenedor Través De La Formació Tecnològica | Upsocial'. N.p., 2015. Web. 28 Maig 2015.
- Wilson, Richard, and Richard Google+. 'Computer Programming Will Soon Reach All Estonian Schoolchildren – Ubuntu Life'. Ubuntu Life. N.p., 2014. Web. 29 Maig 2015.
- Xtec.cat. Currículum provisional 1r i 3r ESO 2015-2016. Proposta curricular de l'ESO. Àmbit científicotecnològic. Web. 28 Maig 2015.
- Xtec.cat,. 'XTEC - Centres. Als Centres... Premis I Concursos. Concurs D'apps 2014'. N.p., 2015. Web. 27 Maig 2015.
- Xtec.cat,. 'XTEC - Currículum I Orientació. Educació Secundària Obligatòria. Optativa d'informàtica a 4rt d'ESO'. N.p., 2015. Web. 27 Maig 2015.